

**UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS**

DATATERN, INC.

Plaintiff

v.

BLAZENT, INC., MICROSTRATEGY INC.,
CARL WARREN AND COMPANY
INCORPORATED, LANCET SOFTWARE
DEVELOPMENT, INC., AIRLINES
REPORTING CORP., MAGIC SOFTWARE
ENTERPRISES LTD., MAGIC SOFTWARE
ENTERPRISES, INC., TERADATA
CORPORATION, INFORMATICA
CORPORATION, EPICOR SOFTWARE
CORPORATION, and PREMIER, INC.

Defendants

C.A. No. 11-cv-11970-FDS

(Consolidated)

**PLAINTIFF DATATERN'S OPPOSITION TO MOTION FOR
SUMMARY JUDGMENT OF NON-INFRINGEMENT**

DATATERN, INC.

By its Attorneys:

William A. Zucker (BBO# 541240)
Lee Carl Bromberg (BBO# 058480)
Erik Paul Belt (BBO# 558620)
Kia Lynn Freeman (BBO# 643467)
Keith Toms (BBO# 663369)
McCARTER & ENGLISH, LLP
265 Franklin Street
Boston, Massachusetts 02110
Tel: (617) 449-6500
Fax: (617) 607-9200

As its own literature confirms, MicroStrategy’s Business Intelligence (“BI”) software platform contains a “unified *object model* to define and construct objects that represent any business.” Ex. B, MicroStrategy Platform Architecture at 17. Indeed, “[t]he MicroStrategy *object model is the genius* of the MicroStrategy platform.” *Id.* (emphasis added). Thus, because its own literature starkly contradicts its contention that it does not have an “object model” — which is the sole basis for the present motion — MicroStrategy’s motion for summary judgment of non-infringement must be denied.

Indeed, while it is cloaked in the guise of summary judgment, MicroStrategy’s motion is in fact a motion for default — the ultimate penalty — as a discovery sanction against DataTern for not inspecting the source code while the case was stayed in 2012. Indeed, MicroStrategy provides no evidence or expert testimony that its product lacks an object model; rather, its entire argument is based on the strange contention that DataTern should not be allowed to conduct *any discovery on infringement* because it opted — as was its choice — to forego a review of the source code for *claim construction purposes* while the case was stayed. MicroStrategy’s faulty logic, however, is not supported by any reasonable interpretation of the Local Rules or the Court’s Order on the Motion to Stay. Thus, sanctions — especially the ultimate sanction of summary judgment — are clearly unwarranted.

BACKGROUND

A. The ‘502 Patent

This case concerns the infringement of DataTern’s U.S. Pat. No. 6,101,502 (“the ‘502 patent”). The ‘502 patent is directed to methods for enabling software applications written in object-oriented code to access data stored in relational databases. In the late 1990s, when the claimed methods were invented, the software industry was experiencing a problem known as the

“object-relational mismatch.” CSoF ¶ 1¹. That is, software written in the most popular form of programming (so-called “object oriented” programming) could not easily access data stored in the most popular form of computer database (the so-called “relational database”) because the organizational structures of each differed. The ‘502 patent solved this problem by introducing, in essence, the use of an intermediary between the application software and database software. CSoF ¶ 4. By using this invention, the database could be changed, and data could be accessed, without requiring the application to be rewritten, thus increasing efficiency. CSoF ¶ 2. The happy result of this innovation is that “neither programmers nor software applications need have knowledge of the database structure, the database programming interface, database security, or the database transaction model in order to obtain access to the relational database.” CSoF ¶ 3. In other words, the ‘502 patent “provides transparent access to the relational database.” CSoF ¶ 2.

The patent summarizes the invention as follows:

In accordance with the present invention, a mapping between an object model and a relational database and a runtime engine are employed to facilitate access to a relational database . . . The database schema, object model, and mapping are employed to provide interface objects that are utilized by an object oriented software application to access the relational database . . . The interface object and runtime engine perform read and write operations on the database.

‘502 Pat. at 1:53–66, CSoF ¶ 2.

The term in dispute in the present motion is “object model.” In a similar way to how a blueprint could show the overall plan for a building, an “object model” represents particular relationships among some of the attributes, objects, and/or classes in an object-oriented application. CSoF ¶¶ 5; Cohen Decl. at ¶ 8. Examples of such relationships that might be captured in an object model include inheritance relationships between classes, object attribute

¹ “CSoF refers to the Counter Statement of Material Fact in Support of Opposition, filed concurrently herewith.

membership, or the presence of references from one kind of object to another. CSof ¶ 5; Cohen Decl. at ¶ 8. Figure 3 of the ‘502 patent provides an example object model:

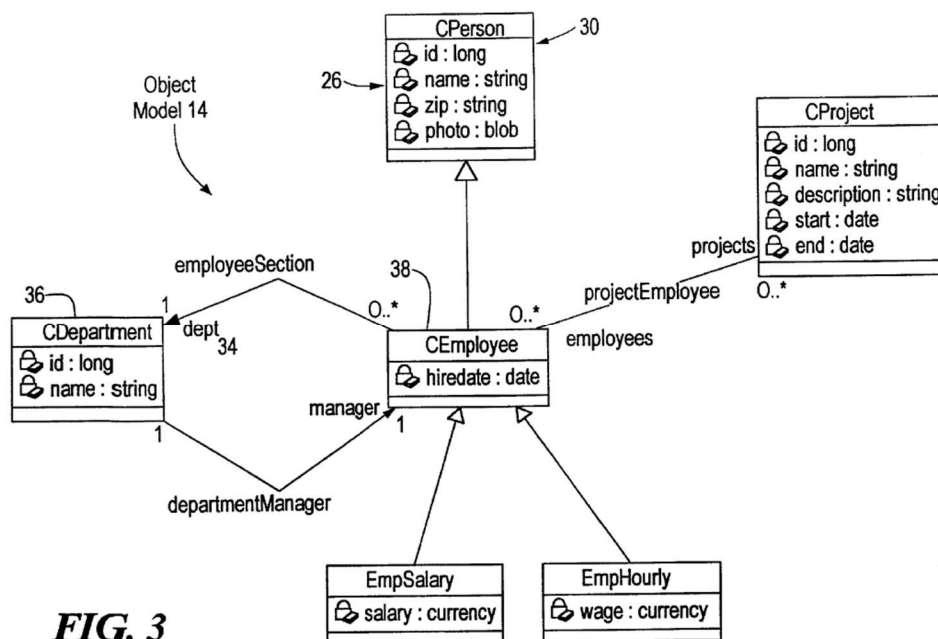


FIG. 3

It shows some features of a set of classes and some relationships between them. CSof ¶ 6; Cohen Decl. at ¶ 8. It does not show any methods or behaviors of the relevant classes. The ‘502 patent also describes the object model as “a template that has a predetermined standardized structure.” CSof ¶ 7. Moreover, the object model “can be created from database schema or database schema can be created from the object model.” CSof ¶ 8.

Claim 1 of the ‘502 patent (with the disputed term **bolded**) provides as follows:

1. A method for interfacing an object oriented software application with a relational database, comprising the steps of:
 - selecting an **object model**;
 - generating a map of at least some relationships between schema in the database and the selected **object model**;
 - employing the map to create at least one interface object associated with an object corresponding to a class associated with the object oriented software application; and

utilizing a runtime engine which invokes said at least one interface object with the object oriented application to access data from the relational database.

‘502 Pat. at 7:53–8:3.

B. MicroStrategy’s Business Intelligence Platform

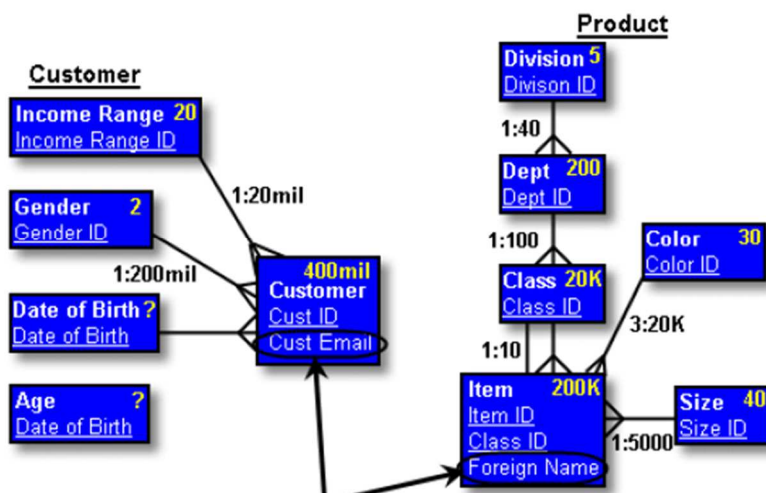
MicroStrategy offers a BI platform, which is an interrelated collection of object-oriented programs that allows its customers to easily access, analyze, and interact with the data in their relational databases. CSof ¶¶ 10, 11. Business intelligence software, in general, allows a company to use data intuitively, in forms “that are clearly named in business terms that everyone from the line manager to the CEO can recognize.” CSof ¶ 10. For example, MicroStrategy’s software allows business people to generate useful reports with intuitive business terms like “customers,” “sales,” or “department,” without having to know anything about programming or where the relevant data is stored in the company’s databases. CSof ¶ 12. Thus, much like the ‘502 patent, MicroStrategy’s software provides “transparent access to the relational database.” See CSof ¶¶ 13; ‘502 Pat. at 1:63–64.

Also like the ‘502 patent, MicroStrategy has an object model that plays a key role in its business intelligence software. CSof ¶¶ 14-26; RSoF² ¶¶ 37, 38, 52-54 (disputing contrary contention). MicroStrategy’s software is based on “a single, unified object model to define and construct objects that represent any business.” CSof ¶ 15. “The MicroStrategy object model is the genius of the MicroStrategy Platform.” CSof ¶ 16. Indeed, as one “Trusted Guru Expert” put it, “MicroStrategy is basically a relationship driven object modeling tool . . .” CSof ¶ 18.

This object model, which is also called the “Logical Data Model,” a “Logical Business Model” or just a “data model,” is comprised of various metadata objects, including the object types that MicroStrategy calls “Facts” and “Attributes.” CSof ¶¶ 19-20. Facts represent

² “RSoF” refers to DataTern’s Responsive Statement of Facts, filed concurrently herewith.

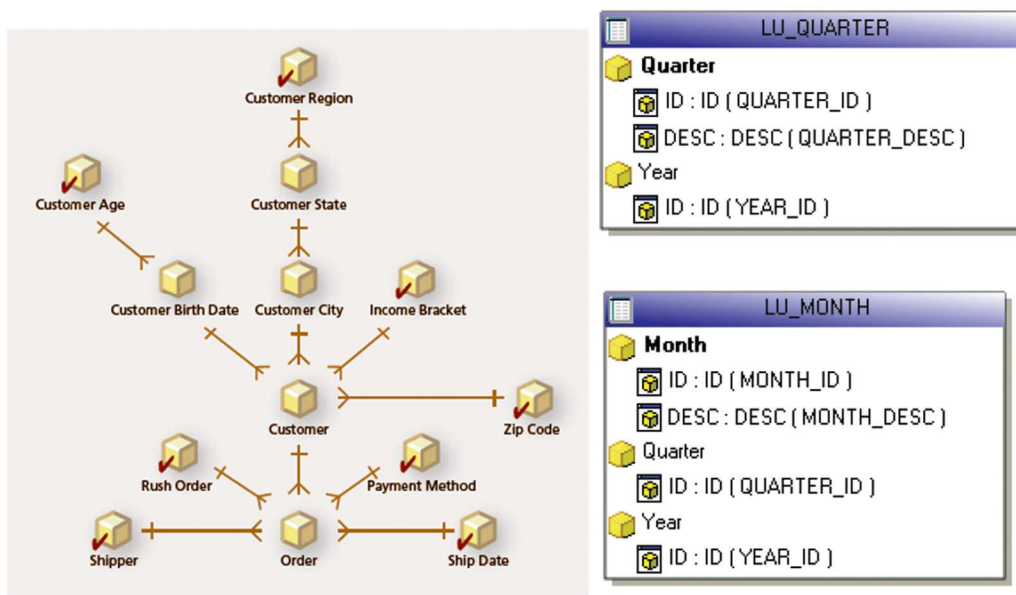
numeric data values, such as number of units sold or revenues. CSof ¶ 21. Attributes represent the context for the data, and include information such as dates, customers, regions, and the like. CSof ¶ 22. The following is a sample Logical Data Model from MicroStrategy's literature that shows how Attributes can be interrelated:




CSof ¶ 24, Ex. C, Design Guide at 34. For example, the “Customer” Attribute includes data for the Customer’s ID and Email address, and a “many-to-one” relationship with each of the “Income Range,” “Gender,” and “Date of Birth” Attributes. *See* CSof ¶ 24.

Notably, MicroStrategy does not use the word “Attribute” in the same sense as the ‘502 patent. CSof ¶ 23. “Attribute” in MicroStrategy’s terminology refers to a type of metadata object, for example, as opposed to a “Fact.” In contrast, “attribute” in the ‘502 patent generally refers to the data component of a class, for example, as opposed to a “method.” *See* Cohen Decl. at ¶ 15 n.1. As such, MicroStrategy Attributes can have attributes. For example the Customer Attribute above has Cust_ID and Cust_Email attributes, which are mapped to columns of the relational database. *Id.*; CSof ¶ 25; Ex. C, Design Guide at 76 (“The ID form of the project attribute must be mapped to the column to relate the two systems of data.”).

Before MicroStrategy's BI platform can be used, however, the customer needs to configure it by programming his or her business's Logical Data Model into the MicroStrategy software. CSof ¶ 27. For example, businesses can use the MicroStrategy Architect tool to "create[] the metadata objects [e.g., facts and attributes] that map the physical structure of a database to a logical, object-oriented model of the business," *i.e.*, the Logical Data Model. CSof ¶ 30; CSof ¶ 29 ("BI Architects use MicroStrategy Architect to: Map the physical database schema into a logical business model with easy-to-use wizards and editors"). The following are sample images from Architect displaying different views of Logical Data Models programmed in Architect:



On the left is the "Hierarchy View," which shows the relationships between various Attribute Objects (which are denoted with the  graphic in both images), such as Customer and Order. See CSof ¶¶ 32. On the right is a view that shows the Attributes and the data items they included as "attributes" (in the object-oriented use of the word). For example, the Quarter Attribute has attributes for an ID and a description, which are mapped to columns in the table. See CSof 32.

MicroStrategy offers a tutorial video where the presenter programs an example Logical Data Model into MicroStrategy and maps that model to the database. *See* CSoF ¶¶ 33-35; Architect Demo *available at* <https://youtu.be/gbj7R1j-sv4>.³ On the “Topics to be Reviewed” slide, the MicroStrategy presenter notes that he is going to be “Mapping Table elements to create MicroStrategy objects” including Attributes and Facts. CSoF ¶ 34. After programming the Logical Data Model into the software, the presenter notes that he can then “use this set of *objects* to create reports.” CSoF ¶ 35 (emphasis added). First, however, he had to update the schema [i.e., the metadata] “just to make sure everything we did is actually saved . . . and all the relationships we created are now *known by MicroStrategy*.” CSoF ¶ 35 (emphasis added).

Thus, contrary to MicroStrategy’s contention, the Logical Data Model does not exist in its customers’ heads.⁴ CSoF ¶¶ 27-35; RSoF ¶¶ 52-55, 59-60; Cohen Decl. at ¶¶ 18–22. As MicroStrategy’s literature confirms, “[t]he MicroStrategy metadata is the manifestation of the object model.” CSoF ¶ 20. The metadata is a “repository that stores MicroStrategy object definitions.” CSoF ¶¶ 31, 49.

In addition, MicroStrategy’s literature makes clear that the object model metadata is “object oriented.” *See* CSoF ¶¶ 36-45, Ex. B, Architecture Guide at 29 (MicroStrategy incorporates “the following architectural design tenets: . . . Dynamic, *object-oriented metadata* objects.”) (emphasis added). For example, the literature states that “the *object-oriented nature* of the metadata repository allows all changes in the logical model to propagate immediately and

³ This YouTube video tutorial is part of the MicroStrategy University series on its official YouTube page. *See* <https://www.youtube.com/user/MicroStrategyBI>.

⁴ MicroStrategy’s argument to the contrary is merely semantic hand wrangling and is without substantive effect. Contrary to its position, MicroStrategy does not employ a rigorous terminology, but instead interchanges many terms, including Logical Data Model, to describe various stages of the object modeling process required to configure MicroStrategy’s software. *See* CSoF ¶ 19.

transparently to all dependent objects.” CSoF ¶ 39 (emphasis added). Likewise, it is apparent from the literature that the object-oriented metadata includes inheritance relationships and methods. *See* CSoF ¶¶ 42-43; Cohen Decl. at ¶¶ 34-38; Ex. B, Architecture Guide at 29 (noting that “every MicroStrategy metadata object contains ‘pointers’ to other metadata objects” so objects are “dynamically assembled from the latest versions of all the other objects contained in the object”).

C. Procedural History

Starting in November, 2011, DataTern sued MicroStrategy and several of its customers, alleging infringement of the ‘502 patent. CSoF ¶ 54. In parallel to the MicroStrategy litigation, DataTern was defending the ‘502 patent against Declaratory Judgment actions filed by Microsoft and SAP in the Southern District of New York (the “New York case”). *See* CSoF ¶ 55.

At the beginning of this case, DataTern suggested to MicroStrategy that the most efficient way to proceed with discovery would be for DataTern to prepare its initial infringement contentions based on a review of MicroStrategy’s source code. CSoF ¶ 56. MicroStrategy refused, however, opting instead to require DataTern to go through the exercise of preparing and serving initial infringement contentions based on publically available materials. *See* CSoF ¶ 57. DataTern did so, and served its Initial Infringement Contentions on April 26, 2012. *See* CSoF ¶ 58-61; RSoF ¶¶ 3-5.

When MicroStrategy objected to the level of specificity of the initial infringement contentions, DataTern once more offered to review MicroStrategy’s source code to provide supplemental contentions based on that review. CSoF ¶ 62. MicroStrategy again refused this reasonable and efficient course of action, and instead moved to compel more specific contentions. CSoF ¶ 63. While the Court granted the motion to compel, Judge Stearns stated “at

this preliminary stage, it is not necessary for plaintiff to provide highly detailed or ultimately successful contentions.” CSoF ¶ 64. Likewise “Plaintiff is not required to know or disclose all possible theories of infringement at this point, and may uncover and disclose additional theories of infringement in the course of discovery.” *Id.*; RSoF ¶ 9.

DataTern went above and beyond what the Court’s Order required by serving highly detailed Amended Infringement Contentions [D.I. 72], which are more than sufficient to disclose the accused products and a theory of infringement. *See* CSoF ¶¶ 65-66. Notably, the Amended Infringement Contentions were still only based on publicly available information because of MicroStrategy’s refusal to disclose source code and also because it had not provided any discovery in the case. CSoF ¶ 66. As such, the Amended Infringement Contentions were not required to be, or understood to be, *final* contentions in any respect. CSoF ¶¶ 76-80; RSoF ¶¶ 18-19.

While MicroStrategy continued to complain about the infringement contentions, it effectively conceded that they were in compliance with the Court’s July 2, 2012 Order and the requirements of Rule 11.⁵ Instead, on August 10, 2012, MicroStrategy unilaterally produced portions of the source code it deemed relevant and demanded that DataTern come and inspect it within 90 days. CSoF ¶ 69. Notably, this production of source code was not pursuant to an agreement of the parties, as MicroStrategy had rejected DataTern’s proposal to inspect the source code previously. *See* CSoF ¶¶ 56, 62, 67. Likewise, a timed inspection was not required by the

⁵ MicroStrategy’s claim that it did not want to “further burden the Court” is a thin excuse for its acquiescence to the sufficiency of the Amended Infringement Contents under the Court’s July 2, 2012 Order.

Scheduling Order. *See* CSoF ¶ 69; Scheduling Order of Mar. 21, 2012; Ex. F, Trans. at 18:24–19:3.⁶

Shortly after MicroStrategy produced the selected source code, however, the Court in the New York action issued its August 24, 2012 claim construction opinion. On reviewing the opinion, DataTern foresaw its ramifications for the MicroStrategy cases. Because the Federal Circuit’s construction would be *stare decisis* for the ‘502 patent, the outcome of that appeal could have been dispositive in this case as well. Thus, DataTern moved to stay the instant case, including discovery and the source code review, pending the final judgment in the New York action. CSoF ¶¶ 72. MicroStrategy opposed.

At the October 5, 2012, hearing on the Motion to Stay, the Court granted the stay. CSoF ¶ 71. The Court, however, stated that it wanted to “hit the ground running” and “proceed promptly to claim construction” should the stay be lifted. *See* CSoF ¶ 72. MicroStrategy, however, pressed the Court to *require* DataTern to review the source code and provide, effectively, *final* infringement contentions that could prove infringement absent any other discovery. *See* CSoF ¶ 74. DataTern objected that such an onerous burden was not required by the Local Rules, the Scheduling Order, or the requirements of Claim Construction. CSoF ¶¶ 73, 75 (MR ZUCKER: “really what is being requested here is [DataTern’s] final expert report. That’s really what they’re requesting. And that was never in the contemplation of the schedule”). DataTern, however, made clear that, while it did not need the source code for claim construction purposes, it would eventually need to review it to prove the ultimate question of

⁶ While the parties’ proposed Rule 26(f) Joint Statement contemplated supplementation of infringement contentions within 90 days of the production of the source code, Judge Stearns never entered the Joint Stipulation, opting instead to set his own Scheduling Order. *See* Scheduling Order, Mar. 21, 2012; CSoF ¶ 68 (“THE COURT: . . . But if the deadline isn’t in the scheduling order, it doesn’t exist as far as I’m concerned. There’s no such thing as private deadlines.”).

infringement. *See* CSoF ¶ 77 (“MR. ZUCKER: I’m not saying, your honor, just so I’m absolutely clear, that at some point in time we don’t need to have access to the source code. . . . the citations to the actual source code would normally be included in the expert reports” during expert discovery).

The Court did not accept MicroStrategy’s invitation to obligate DataTern to review the source code and provide further amended infringement contentions during the stay or before claim construction. CSoF ¶¶ 76, 78; RSoF ¶¶ 17–19. Not having the infringement contentions in front of it, the Court declined to address their sufficiency. CSoF ¶ 79. The Court, however, permitted DataTern to review the source code if it reconsidered its need to do so for *claim construction purposes*. CSoF ¶ 80 (“I will permit DataTern, if it chooses, to inspect the source code now. It says it doesn’t need to [i.e., for claim construction purposes],⁷ but I would not view that as a violation of the stay.”).

Final judgment issued in the New York case on December 26, 2012, and DataTern appealed the New York court’s construction of three terms of the ‘502 patent, “object model,” “to create at least one interface object,” and “runtime engine.” In addressing the appeal, however, the Federal Circuit construed only the term “object model.” While the court upheld the judgment based on the requirement that “object model” have classes, it also found that the classes of the ‘502 patent need not have behaviors and required only attributes. *See* CSoF ¶ 84;

⁷ It is DataTern’s reasonable understanding that the Court was referring to a source code review for claim construction when it made the statement: “if you think you need to [review the source code] the time to do it is now. If you think you don’t need to do it, we’ll take that up in due course.” CSoF ¶ 80. First, this was in response to counsel for MicroStrategy asking the Court if it agreed that “supplemental infringement contentions citing source code are required and, *in particular, are required before the claim construction process proceeds.*” CSoF ¶ 78 (emphasis added). Second, the reference to DataTern thinking that it did not need to review the source code clearly refers back to DataTern’s position that it did not need the source code for claim construction purposes, though it would need to see the source code eventually for its ultimate case on the merits. CSoF ¶ 77.

RSof ¶ 36. The Court also declined to reach the other claim construction issues for this term, including whether an “object model” requires inheritance relationships or whether it needs to be the object model of the object oriented software application. *See* CSof ¶ 84.

This case was stayed until final judgment issued in the New York case (prior to appeal), at which time DataTern conceded non-infringement based on the New York court’s construction of “to create at least one interface object.” The Court entered judgment on the basis on that concession. On the appeal of the instant case, the Federal Circuit reversed the New York court’s construction and remanded the case for further proceedings. CSof ¶¶ 81-83.

On remand, rather than “hit the ground running” for claim construction, MicroStrategy moved for Summary Judgment. While the present motion is styled as a motion for Summary Judgment based of the Federal Circuit’s construction of “object model,” it is really a motion for discovery sanctions attempting to default DataTern for not reviewing the source code during the stay. Indeed, MicroStrategy does not offer any facts or expert testimony to support the position that it does not have an “object model.” Rather, its sole argument is that DataTern cannot prove infringement because it has allegedly lost its one and only shot to review the source code *for all purposes*, not just for the purpose of claim construction. This frivolous allegation, however, has no basis in the record, and is just a further example of MicroStrategy’s campaign of gamesmanship over infringement contentions.

Notably, only contention discovery was completed before the stay. Other than its gamesmanship regarding the source code production, MicroStrategy has not provided a scrap of fact discovery relevant to the issue of infringement. CSof ¶ 66. It has produced no documents, it has provided no interrogatory answers, and no depositions have been taken.

ARGUMENT

Because its own product literature describes “[t]he MicroStrategy *object model* [as] the *genius* of the MicroStrategy platform,” the evidence of record, when taken in the light most favorable to DataTern, requires that the motion for summary judgment be denied. Ex. B, Architecture Guide at 17. Indeed, in view of the blatant admissions in its literature, and its failure to offer an expert declaration in support of its untenable position, MicroStrategy should never have brought this motion in the first place.

Nonetheless, despite its styling, MicroStrategy’s motion is not really a motion for summary judgment at all. Rather, MicroStrategy implicitly and improperly seeks the extreme sanction of precluding DataTern from taking any discovery in the case — including, but not just limited to, a source code review — to prepare its infringement case for trial. DataTern, however, has violated no Local Rule or Court Order, and thus sanctions of any sort would be wholly inappropriate, let alone the terminal sanctions MicroStrategy seeks. In sum, while it is dressed in the garb of summary judgment, MicroStrategy’s motion is just another gambit in its vexatious pattern of its infringement contention gamesmanship.

As the moving party, MicroStrategy bears the ultimate burden of demonstrating that “no fact that is material to disposition of the case is genuinely in dispute.” *Cohesive Technologies v. Waters Corp.*, 130 F. Supp. 2d 157, 160 (D. Mass. 2001). Only if MicroStrategy submits *prima facie* evidence that satisfies that burden must DataTern come forward with evidence showing a genuine issue of material fact. *See id.*; *Saab Cars USA, Inc. v. United States*, 434 F.3d 1359, 1368–69 (Fed. Cir. 2006) (affirming denial of summary judgment because non-moving party not required to submit opposing evidence if moving party’s evidence inadequate); *Hilgraeve Corp. v. McAfee Associates, Inc.*, 224 F.3d 1349, 1352 (Fed. Cir. 2000) (vacating summary judgment of

no infringement due to remaining fact issues). On summary judgment, this Court must view all facts in a light more favorable to the opposing party, DataTern, and draw all inferences in DataTern's favor. *Absolute Software, Inc. v. Stealth Signal, Inc.*, 659 F.3d 1121, 1133 (Fed. Cir. 2011) (vacating summary judgment of non-infringement because "The decision fails to draw a reasonable inference in favor of Absolute, the non-moving party"); *O'Connor v. Steeves*, 994 F.2d 905, 907 (1st Cir. 1993) (vacating summary judgment because "The court must view the entire record in the light most favorable to the non-moving party and indulge all reasonable inferences in that party's favor.")

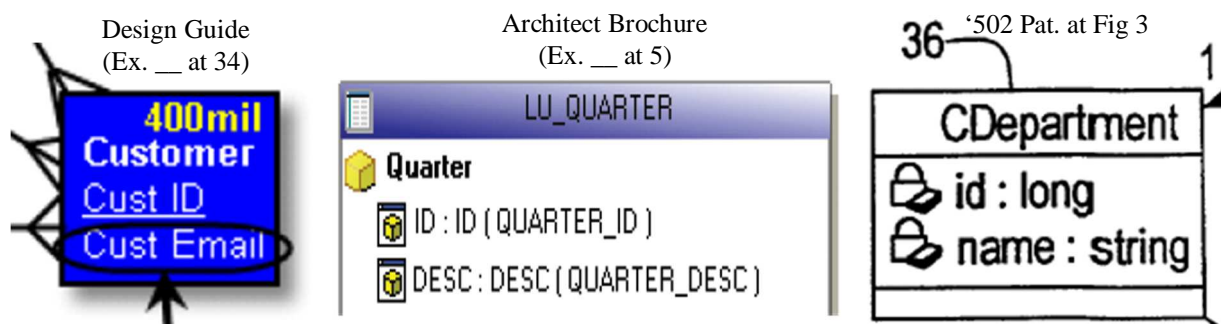
I. MICROSTRATEGY HAS AN OBJECT MODEL WITH CLASSES

Summary Judgment must be denied because the evidence, taken in the light most favorable to DataTern, shows that MicroStrategy's business intelligence platform has an "object model." RSOF ¶¶ 37-38, 58-60. Indeed, as its own Architecture Guide confirms, not only does MicroStrategy have "a single, unified *object model* to *define and construct* objects that represent any business," it specifically describes the object model as the "*genius* of the MicroStrategy platform." Ex. B, Architecture Guide at 17. Given these stark admissions — which MicroStrategy offers no testimony to rebut — it is clear that there is, at the very least, a material issue of disputed fact, and thus Summary Judgment must be denied. Indeed, given its total failure to offer any evidence of the lack of an object model, MicroStrategy fails to even meet its *prima facie* burden as the movant for summary judgment. *See Saab*, 434 F.3d at 1368–69.

The "object model" discussed in the Architecture Guide includes the Logical Data Model identified in the Amended Infringement Contentions. Contrary to MicroStrategy's odd contention, the Logical Data Model is not in the customers' head; it is actually contained in the MicroStrategy platform. *See* Cohen Decl. at ¶¶ 18–22; RSOF ¶¶ 52–55, 59–60. MicroStrategy's software cannot read its customer's minds, so it offers tools, such as MicroStrategy Architect,

which allow its customers to create “a representation of [a] organization’s business as MicroStrategy objects in the metadata.” Ex. D, Architect Brochure at 4. Indeed, as the MicroStrategy University instruction video shows, users of MicroStrategy must create and map an object model to their data stores before the software can be used. *See* CSof ¶¶ 32-35.

Also, the evidence of record, when viewed in the light most favorable to DataTern, establishes that the MicroStrategy object model has “classes.” *See* CSof ¶¶ 46-53; Cohen Decl. at ¶¶ 23-38. A class is a definition or template of an object, which includes attributes, and which can also have other characteristics, such as inheritance relationships or methods. *See* CSof ¶¶ 47-48; Cohen Decl. ¶ 5-7. MicroStrategy’s object model fits this bill. First, it is used to “define and construct objects that represent any business.” Ex. B, Architecture Guide at 17; Ex. B, Design Guide at 8, 10 (the metadata is a “repository that stores MicroStrategy object definitions.”); Ex. C, Design Guide at 33-34 (“Each element of the Customer attribute represents a different customer.”). In addition, like the ‘502 patent, the metadata objects have attributes, for example:



Cohen Decl. at ¶¶ 31-33. Although inherence relationships and methods are not required by the Federal Circuit’s construction of “object model,” those too are also present in MicroStrategy’s product. *See* Cohen Decl. at ¶¶ 34-38.

In sum, the record in the light most favorable to DataTern shows that MicroStrategy’s software has a “single, unified object model” that is “manifest” in a “repository” of “object-

oriented metadata” which stores “object definitions” that include attributes, methods, and inheritance relationships — in other words, it is an object model with classes. CSoF ¶¶ 46-53; Cohen Decl. at ¶ 39. Like the proverbial rose, a class by any other name is still class. Thus, MicroStrategy’s business intelligence platform has the “object model” of the ‘502 patent, and therefore summary judgment of non-infringement must be denied.

II. DATATERN SHOULD BE ALLOWED DISCOVERY, INCLUDING SOURCE CODE REVIEW, BEFORE FINAL INFRINGEMENT CONTENTIONS

A. MicroStrategy’s Request for Discovery Sanctions Must be Denied

DataTern has violated no Federal Rule, Local Rule, or Order of this Court, and thus the severe evidentiary sanctions MicroStrategy seeks — which effectively amount to dismissal of the case — are unwarranted and wholly inappropriate. Pursuant to the Stay hearing, DataTern was under no obligation to review the source code and prepare *final* infringement contentions with citations thereto, and thus has not in any way waived its right to review the source code or supplement its contentions now that the case is remanded. RSoF ¶¶ 17–19. Indeed, DataTern sought a stay specifically to avoid the expense of potentially unnecessary discovery, including preparation of final contentions and source code review, in light of the (now reversed) construction of “to create an interface object.” Thus, MicroStrategy’s motion for sanctions must be denied.

As this Court has noted, among local patent rules, “Local Rule 16.6 . . . is one of the less onerous rules concerning preliminary infringement contentions.” *Trustees of Boston University v. Everlight Electronics Co., Ltd.*, 12-cv-11935, 2013 WL 2932822, at *3 (D. Mass. June 12, 2013) (Saylor, D.J.). L.R. 16.6, which requires the parties to discuss “the timing for disclosing initial infringement and invalidity positions,” lacks the hard deadlines, detailed disclosure mandates, strict limits on amendments, and other practices that are features of other courts’ local

rules. Thus, as the Court has noted, “the interpretations of different local rules by different district courts has *no binding or persuasive* influence on this Court’s interpretation of its own local rules.” *Id.* (emphasis added) (declining to consider initial disclosure precedent based on other district’s local rules). Further, Courts in this jurisdiction show heavy reluctance to impose terminal sanctions for what amounts to a discovery dispute. *See, e.g., Benitez-Garcia v. Gonzalez-Vega*, 468 F.3d 1 (1st Cir. 2006) (“it has long been our rule that a case should not be dismissed with prejudice except when a plaintiff’s misconduct is particularly egregious or extreme.”).

Nothing in the Local Rules, the Scheduling Order, or other Orders of this Court obligated DataTern to review the source code and prepare final infringement contentions during the stay or prior to claim construction. *See* Ex. F, Trans. at 32:24–33:3 (permitting, but not requiring, DataTern to inspect the source code for claim construction purposes). Indeed, the Court expressly declined MicroStrategy’s invitation to issue just such an Order at the hearing on the motion to stay. *See id.* at 36:10–37:12. DataTern simply cannot be sanctioned for not following an order that MicroStrategy failed to obtain.

Likewise, that the Court allowed DataTern the option of reviewing the source code during the stay *if needed for claim construction* will not support the draconian sanctions that MicroStrategy seeks. DataTern has always contended that the review of source code — while important for the ultimate proof of infringement at trial — was unnecessary to satisfy the notice function of preliminary infringement contentions for claim construction purposes. *See* Ex. F, Trans at 17:6–18:7. Foregoing an expensive and unnecessary review of the source code for claim construction *while the case was stayed*, however, in no way waives or forecloses DataTern’s right to review the source code in the ordinary course of discovery (which had yet to

begin when the case was stayed). Likewise, allowing source code review during the discovery, when reopened, will in no way slow or impede the progress of the case.

Contrary to MicroStrategy's contention, the Court was referring to the optional source code review for claim construction purposes when it said, "if you think you need to do it, the time to do it is now. If you don't think you need to do it, we'll take that up in due course." *See* Ex. F, Trans at 37:5–11; RSoF ¶ 19. Not only had the Court previously expressed the desire to "hit the ground running" with claim construction, the Court was also responding to a question regarding what was required "before the claim construction process proceeds." Ex. F, Trans. at 36:10–15. Likewise, the Court's statement is a clear reference to the exchange with DataTern, who took the position that the Amended Infringement Contentions are adequate for the purposes of claim construction, but that a review of source code would be appropriate later in the case. *See* Ex. F, Trans. at 17:6–18:7. Notably, MicroStrategy does not even attempt to challenge the sufficiency of the Amended Infringement Contentions for the purposes of claim construction.

DataTern cannot be sanctioned — let alone effectively defaulted — for not following the local rules of *some other court*. MicroStrategy cites no cases from this jurisdiction applying L.R. 16-6 to *require* — let alone sanctioning the absence of — final infringement contentions before claim construction. Indeed, the cases that MicroStrategy cites all apply the Local Rules from other jurisdictions or court orders specific to the parties therein, and thus have no authority here.⁸ *See Trustees of Boston University*, 2013 WL 2932822, at *3.

⁸ *Sloan Valve Co. v. Zurn Indus., Inc.*, No. 10-CV-00204, 2014 WL 51293, at *3 (N.D. Ill. Jan. 7, 2014) (denying motion to amend contentions after deadline imposed expressly by N.D. Ill. local rules); *St. Clair Intellectual Prop. Consultants, Inc. v. Matsushita Elec. Indus. Co.*, No. CA 04-1436-LPS, 2012 WL 1015993, at *1 (D. Del. Mar. 26, 2012) (denying leave to amend contentions after deadline imposed by Rule 16 Scheduling Order); *Acer, Inc. v. Tech. Properties Ltd.*, No. 5:08-CV-00877 JF/HRL, 2010 WL 3618687, at *3 (N.D. Cal. Sept. 10, 2010) (denying leave to amend contentions where N.D. Cal. local rules permitted amendment

MicroStrategy's reliance on *St. Clair Intellectual Prop. Consultants, Inc. v. Matsushita Elec. Indus. Co.*, No. CA 04-1436-LPS, 2012 WL 1015993, at *1 (D. Del. Mar. 26, 2012), in attempting to fault DataTern for not anticipating the Federal Circuit's decision, is particularly misplaced. Not only was that case not applying L.R. 16-6, it also involved a request to supplement *expert reports* to add a new infringement theory after the close of expert discovery, and thus has no applicability where infringement fact discovery has not even started in earnest. *Id.* at *St. Clair*, 2012 WL 1015993 at *5.

B. DataTern Is Entitled to Amend Its Infringement Contentions

There is nothing in the Local Rules or the Orders of the Court that forecloses DataTern from amending its Infringement Contentions to supplement its infringement theories as the discovery process continues. To date, DataTern has received no discovery from MicroStrategy on infringement, and as more information is produced, DataTern will find itself in a position to provide even more support for its infringement theories. No party is expected to be able to produce final Infringement Contentions — let alone *prove infringement for summary judgment* — before discovery has taken place. *See* Order July 2, 2012 (“Plaintiff is not required to know or disclose all possible theories of infringement at this point, and may uncover and disclose additional theories of infringement in the course of discovery”); *Trustees of Boston Univ.*, 2013 WL 2932822, at *4 (“At this early stage, before fact discovery has been completed, such infringement contentions are specific enough to establish the existence of a good faith basis as required by Rule 11”).

only by court order); *Nike, Inc. v. Adidas Am. Inc.*, 479 F. Supp. 2d 664, 667 (E.D. Tex. 2007) (striking amended contentions for untimeliness under E.D. Tex. local patent rule 3”); *Convolve, Inc. v. Compaq Computer Corp.*, No. 00CIV.5141(GBD)(JCF), 2006 WL 2527773, at *3-4 (S.D.N.Y. Aug. 31, 2006) (striking invalidity contentions for failure to comply with parties’ Stipulated Order for case).

Moreover, DataTern's Amended Infringement Contentions are sufficiently detailed to move forward with discovery and claim construction, as the Court contemplated at the Stay hearing. As noted above, MicroStrategy does not challenge the appropriateness of the contentions for claim construction, and even its attempt to manufacture an issue regarding the object model contentions is misplaced because it is clear from any reasonable reading of the contentions that DataTern is accusing the Logical Data Model that is programmed into the software, not the data model in the user's head. *See* Amended infringement Contentions [D.I. 72-1] at 4–5 (identifying the Architect functionality to create a Logical Data Model); Order of July 2, 2012 (“at this preliminary stage, it is not necessary for plaintiff to provide highly detailed or ultimately successful contentions”).

In sum, the Court should not reject MicroStrategy's campaign of infringement contention-based gamesmanship and efforts to evade the merits of the case.

CONCLUSION

Because MicroStrategy's software has a “single, unified *object model*” that satisfies the “object model” limitation of the ‘502 patent, and for the other reasons stated above, MicroStrategy's motion for summary judgment and its implicit request for unwarranted sanctions must be denied.

DATATERN, INC.
By its Attorneys:

/s/ Erik Paul Belt
William A. Zucker (BBO# 541240)
Lee Carl Bromberg (BBO# 058480)
Erik Paul Belt (BBO# 558620)
Kia Lynn Freeman (BBO# 643467)
Keith E. Toms (BBO# 663369)
McCARTER & ENGLISH, LLP
265 Franklin Street
Boston, Massachusetts 02110
Tel: (617) 449-6500
Fax: (617) 607-9200
Email: wzucker@mccarter.com
Email: lbromberg@mccarter.com
Email: ebelt@mccarter.com
Email: kfreeman@mccarter.com
Email: ktoms@mccarter.com

April 16, 2015

CERTIFICATE OF SERVICE

I, Erik Paul Belt, hereby certify that on this 16th day of April, 2015, the within document was electronically filed with the Clerk of the Court using the CM/ECF system and will be sent electronically to the registered participants as identified on the Notice of Electronic Filing (NEF), pursuant to Local Rule 5.4(C).

/s/ Erik Paul Belt
Erik Paul Belt